Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 19 (2015) 135 – 140

Procedia Technology

8th International Conference Interdisciplinarity in Engineering, INTER-ENG 2014, 9-10 October 2014, Tirgu-Mures, Romania

Drill Hole Sets Manufacturing on CNC Machines

Mocian Ioan*, Cazacu Razvan

"Petru Maior" University of Tirgu-Mures, Nicolae Iorga 1, Tirgu-Mures 540088, Romania

Abstract

This research proposes an optimized methodology to generating the programs needed for the manufacturing of parts with numerous drill holes on drilling and milling machines. The authors present this new approach and an application that implements it. The application and its accessible user interface is powerful and easy to use by the CNC programmers. The data is modeled in a structure that makes it natural for the engineer to handle while allowing the construction of automation algorithms at the same time. Initial data can be supplied directly inside the application GUI or imported from an exterior spreadsheet file produced by applications such as Excel or generated using code by other applications such as AutoCAD. The application can be extended with any number of postprocessors in order to output specific program syntax for any machine while keeping the core preprocessor and processor intact.

Peer-review under responsibility of "Petru Maior" University of Tirgu Mures, Faculty of Engineering

Keywords: pockets; drill hole sets; CAM; CNC.

1. Introduction

CNC machines programming is a common activity in both large and small mechanical engineering enterprises. Increasing productivity is a major concern even for the companies with a small number of CNC machines. A distinct category of programs are those aimed at the manufacturing of drill holes, usually grouped in sets. Commercial CAD/CAM software packages offer solutions for this type of CNC programs but these have the disadvantages of being too complex and expensive for the aims and possibilities of small companies. Original light-weight and

* Corresponding author. Tel.: -; fax: -.
E-mail address: ioan.mocian@ing.upm.ro

Peer-review under responsibility of "Petru Maior" University of Tirgu Mures, Faculty of Engineering
doi:10.1016/j.protcy.2015.02.020

dedicated solutions are more fitted for these enterprises [1-3]. These solutions can take into account very specific details such as the influence of tool flexibility on manufacturing precision [4] or can offer inside integrated programming environments functionality to automate on knowledge-based principles certain tasks traditionally done by the programmer [5].

CNC drill holes manufacturing programs present several characteristics that require a specific approach. Firstly, one needs the coordinates of all the holes belonging to the manufactured part. These can be specified in a text file, read from the part blueprints or extracted from a geometric model using a dedicated software solution [6].

Regardless of the method used to obtain the hole centers coordinates, this information needs further processing, either performed entirely by the technologist or with the aid of built-in algorithms. At the heart of the manufacturing process lies the concept of “sets”. Several sets of holes need to be constructed in such a way that all holes in the set can be processed by the same tool. A hole usually belongs to more sets, reflecting the technological steps involved in manufacturing it. For example, one must consider that in order to avoid the occurrence of high milling forces, in general a hole with the diameter greater than 8 mm needs to be drilled several times with drills of increasing diameters, up to a size close to the final one. A set is processed with the same drill, after the operation some of the resulting holes being in their final state while some require further processing.

Each hole and set belongs to a category, a manufacturing step performed with a specific tool in specific conditions. Each category is subjected to a number of rules imposing limitations and constraints to the allocation of holes to sets, including mandatory or forbidden precedence rules. The collection of categories is dynamic, allowing the creation of new categories along with custom rules. At its core however, the system contains four categories out of which three represent the typical final product drill holes (the fourth being an initial mandatory centering operation):

- Regular holes – without special requirements, obtained with drills of the same diameter as the hole.
- Precision holes – have specifications regarding diameter tolerance and surface quality. The final drill is performed with the reamer, requiring a small end clearance (0.1 mm - 0.3 mm).
- Threaded holes – finished with a tap, but needs regular drilling to the corresponding diameter.

The CNC equipment can be extended with new functions (new machine codes) callable from the main program [7]. At the same time, some CNC machines contain native macro commands in the numerical control equipment. These subprograms are actually sequences of CNC words that perform several operations with only one command inside the main program [8]. Examples of such macros are:

- Step drilling with tool retraction;
- Delayed drilling;
- Drilling with tool retraction back to the safety plane;
- Threading;
- etc.

2. Data structure and processing

An application for the computer-aided generation of hole sets manufacturing programs on CNC machines involves three phases: initial data input, data processing and outputting actual CNC programs - sequences of words with a syntax compliant to specific machines. In order to facilitate this workflow, the data needs to be modeled in an appropriate manner, as emphasized in [9].

The following sections present the two most important data structures of the system: hole data and sets data - without detailing other components such as categories, rules or postprocessor data. All data inside the system has been modeled on the principles of relational databases, while the functionality relies on the object-oriented programming (OOP) paradigm.

2.1. Hole data

All data regarding the holes is stored inside a collection (named "Holes") of instances of a dedicated data structure (named "Hole"). The data structure contains geometrical information - inputted with respect to a specified coordinate system - and other non-geometrical specifications. Table 1 presents in tabular form an example of hole data, with sample values for the most important fields.

Table 1. Hole data main fields.

No.	X	Y	Z	Depth	Type	Diameter	Observations
1	25.4	32.5	0	10	R	5	Regular hole, +/- 0.1
2	84.2	32.5	0	10	T	6	Metric thread
3	65.4	32.5	-5	5	P	10	Precision hole
…	…	…	…	…	…	…	…

The notations used in Table 1 are explained below:

- X, Y – plane coordinates of the hole;
- Z – the altitude of the upper limit of the hole. The reference coordinate system is assumed to have the origin at the top of the work piece, making the Z coordinate zero or negative. In practice, in order to avoid user input errors Z is treated as a positive parameter, the sign being resolved in the postprocessing phase;
- Depth – actual depth of hole (starting from Z);
- Type – type of hole, belonging to one of the categories defined. In the example above: R – regular hole, with no specific exigences; T – thredead hole; P – precision hole, finished with a reamer.

Hole information represents the initial data of the process. Starting from it, the programmer technologist, assisted by the program, needs to group the hole drilling operations in sets of identical properties. Each hole in a set is processes by the same tool in a single step.

2.2. Sets data

We define a set as the collection of holes that can be processed with the same tool and the same milling parameters. As a consequence each set has a specific tool attached: drill, tap, reamer. Starting from initial data, the CNC programmer prepares the sets to be processed in chronological order, in accordance with the designed technological operations. This activity is aided by the program. The application proposes the sets of holes based on the rules governing the categories to which each hole belongs but the user has full control over the set-up and can edit it in any way as long as the final result is in accordance with the rules.

The first set is always dedicated to the centering operation (performed with a dedicated tool - the center drill) and is composed of all the holes in the job. This is due to the fact that all other operations have a mandatory precedence rule on the centering operation. The configuration of the next sets depends on the chosen technology, according to the initial data specified for the holes: type of hole (regular, precision, thread, etc.), diameter, etc.

Because each hole will probably be part of several sets, the relation between these two concepts is not one of inclusion. Instead, the sets hold references (handles) to the holes they contain, while the holes also retain handles to all the sets they belong to. Running through all the sets referenced by a hole reflects all the technological operations involved in the manufacturing of that specific hole, from centering to the end result.

Table 2. Sets main data fields.

No.	Category	G-code	Tool	Diameter	Rot. speed	Feed rate	Holes	Observations
1	Centering	G81	T1	-	800	60	1-23	All holes
2	Drilling	G82	T6	5	600	50	1,2,6	-
…	…	…	…	…	…	…	…	…

Table 2 presents the main fields of the “Set” data structure in tabular form while also showing sample data for two fictional sets. The sets are organized in a collection (named “Sets”). The whole table can be viewed as a representation of the “Sets” object, while each line represents a “Set” instance.

The most important field is the “Category”, which imposes limitations to the holes that can be added to each set (“Holes” filed). The others are mostly to specify manufacturing parameters. It can be observed that the tools are not labeled consecutively. This is due to the fact that the tool storage is usually pre-organized and its configuration rarely modified. The technologist needs to know the position of each tool inside the tool storage.

After all hole sets are correctly and completely configured (all holes must be assigned to enough sets to make them manufacturable, all rules are enforced, all manufacturing parameters are set, etc.) the next step of postprocessing follows. Depending on the available postprocessors supplied, the technologist has the option to output programs targeted at specific machines, each with its own syntax.

3. The application: GSG-01

The methodology proposed here and described above is currently being implemented in an application called “GSG-01”. The application was designed with a full OOP approach and is developed using Visual Basic .NET [9]. The application is intended to assist the technologist in creating programs for the machining of drill holes in a timely and safely manner. The streamlined user interface and power of the underlying system and data model make the job of the engineer much easier, allowing fast creation of error-free CNC programs.

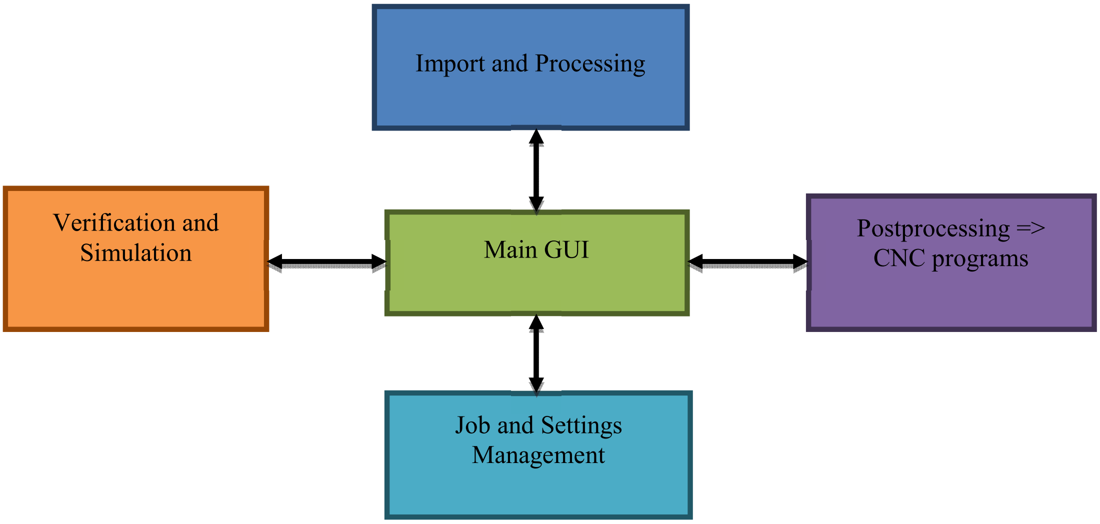

Fig. 1. Application schematic diagram.

As the schematic diagram in Fig. 1 shows, the application consists of several intertwined components, all accessed from the graphical user interface (Main GUI). These components are discussed below:

Import and Processing – handles initial data importing or editing and the configuration of sets. Initial data can be imported in tabular from an external spreasheet. It can also be edited or created directly inside the main application GUI with the help of an interactive workspace. The source of data for the spreadsheet or for direct input can be any dimensioned blueprint or digital model file. Some data regarding each hole can be partialy omitted but only completely configured holes can be added to sets. The processing part consists in the assisted creation of hole sets according to imposed rules. The program can automatically generate a solution for the manufacturing of existing holes, but the user has full control over the final result. Completely manually generated solutions are also possible.

Job and Settings Management – deals with the saving and opening of jobs, the control of job-level settings, as well as with the management of auxiliary data such as the categories collection and their rules. The set of default categories is saved separately and doesn't normally need to be adjusted (it is assumed most holes can be classified as regular, precise or threaded and all need a centering first). However, the users have full flexibility and can add or edit categories, whether these represent actual final holes or just preparatory or intermediate operations (like centering). Job settings handle the properties specific to each manufacturing task as a whole:

- *Safety plane*: the plane above which the tool can move freely without collision hazard; the tool will always get back to this plane after machining a hole and then move to the X-Y coordinates of the next hole;
- *Control point*: a point inside the manufacturing space to which the tool gets moved in order to visually verify that the part is well positioned on the machine table;
- *End program point*: the point where the tool retires after finishing its job so that it can be easily change with the next tool required by the program;

Verification and Simulation – encompasses the data integrity management system (ensured by the data model), and the visual simulation of the current state of the job (with the aid of the interactive graphic workspace). The verification is always performed in real time, meaning the users are always given the information on the completeness and correctness of the data they handle and certain actions that would cause data inconsistency are forbidden. For example, all holes need to have all parameters specified before it can be added to a set. Also, a hole belonging to one or more sets can't have its properties altered so that would make it incompatible with these. It would have to be removed from the conflicting sets first.

Regarding the simulation, the entire problem space is visualized in the main GUI area. The holes can be viewed, created, queried and modified directly inside this visual workspace. The state of the holes - completely defined, incomplete, part of a set, completely configured, etc. - is expressed with the aid of a color code according to which each color corresponds to a certain state. This makes the visual verification of the current job easy and convenient.

Postprocessing – actually a component comprised of several modules. Each targeted machine gets its own postprocessor. The application outputs the actual CNC program as a text file with the syntax specific to the required machine. Once a job is configured, it can be output in any number of available formats.

4. Conclusions and further research

The methodology and application presented in this paper have a strong applicative character, allowing the simplification of program development for small companies possessing CNC equipment but less qualified personnel using it. Considering the tight deadlines often imposed in a concurrency market, the ability to quickly and easily generate error-free programs brings a great advantage to companies benefiting from such software. Another great advantage is the clean and streamlined behaviour of the application, which allows companies to quickly train new engineers in how to use it.

The great power of the application consists of its flexibility and the algorithm for automate hole sets generation which greatly simplifies the manufacturing design process while still leaving the technologists the liberty to fully control the manufacturing process. For more versatility, the application can be enriched with as many postprocessors as needed. The output of the program in new syntax specific to any machine the users might own can be easily achieved with relatively little effort.

References

[1] Mansour S. Automatic generation of part programs for milling sculptured surfaces. Journal of Materials Processing Technology 2002; 127(1):31-39.

[2] Mocian I, Cazacu R. Software for computer aided manufacturing with the MORI SEIKI CNC milling machine. Procedia Technology 2013; 12):369–374.

[3] Cazacu R, Mocian I. Software for computer aided manufacturing of complex pockets. Procedia Technology 2013; 12):364–368.

[4] Hinds BK, Ong TS. End milling of circular pockets to meet geometric tolerances. Journal of Materials Processing Technology 2004; 152(3):339-345.

[5] Zhang S, Wang G, Zhang L, Fang X. CNC programming system for complex components based on KBE within integrated environment of CAD/CAPP/CAM. Frontiers of Mechanical Engineering in China 2009; 4(1):97-102.

[6] Mocian I, Cazacu R. New advances in application development for AutoCAD environment. Proceedings of the 9-th International Conference MTeM 2009; 1:181-186.
[7] Omirou SL, Barouni AK. Integration of new programming capabilities into a CNC milling system. Robotics and Computer-Integrated Manufacturing 2005; 21(6):518-527.
[8] Mocian I. Computer aided technological design in mechanical engineering ("Proiectare tehnologica asistata de calculator în constructia de masini"). Targu Mures: Petru Maior University Publishing House; 1999.
[9] Suh SH, Cheon SU. A Framework for an Intelligent CNC and Data Model. The International Journal of Advanced Manufacturing Technology 2002; 19(10):727-735.
[10] ***, VB.NET Documentation website (online), available online: #http://msdn.microsoft.com/en-us/library/2x7h1hfk%28v=vs.90%29#, accessed: 2014